September 1988

Software Serial Port Implemented with the PCA

BETSY JONES
ECO APPLICATIONS ENGINEER

Order Number: 270531-002

Intel Corporation makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this document nor does it make a commitment to update the information contained herein.

Intel retains the right to make changes to these specifications at any time, without notice.

Contact your local sales office to obtain the latest specifications before placing your order.

The following are trademarks of Intel Corporation and may only be used to identify Intel products:

> 376, 386, 387, 486, 4-SITE, Above, ACE51, ACE96, ACE186, ACE196, ACE960, BITBUS, COMMputer, CREDIT, Data Pipeline, ETOX, Genius, i, i486, i860, ICE, iCEL, ICEVIEW, iCS, iDBP, iDIS, I^{2}ICE, iLBX, iMDDX, iMMX, Inboard, Insite, Intel, int$_e$l, Intel386, int$_e$lBOS, Intel Certified, Intelevision, int$_e$ligent Identifier, int$_e$ligent Programming, Intellec, Intellink, iOSP, iPDS, iPSC, iRMK, iRMX, iSBC, iSBX, iSDM, iSXM, Library Manager, MAPNET, MCS, Megachassis, MICROMAINFRAME, MULTIBUS, MULTICHANNEL, MULTIMODULE, MultiSERVER, ONCE, OpenNET, OTP, PROMPT, Promware, QUEST, QueX, Quick-Erase, Quick-Pulse Programming, Ripplemode, RMX/80, RUPI, Seamless, SLD, SugarCube, UPI, and VLSiCEL, and the combination of ICE, iCS, iRMX, iSBC, iSBX, iSXM, MCS, or UPI and a numerical suffix.

MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

MULTIBUS is a patented Intel bus.

CHMOS and HMOS are patented processes of Intel Corp.

Intel Corporation and Intel's FASTPATH are not affiliated with Kinetics, a division of Excelan, Inc. or its FASTPATH trademark or products.

Additional copies of this manual or other Intel literature may be obtained from:

> Intel Corporation
> Literature Sales
> P.O. Box 58130
> Santa Clara, CA 95052-8130

CG-051989

For microcontroller applications which require more than one serial port, the 83C51FA Programmable Counter Array (PCA) can implement additional half-duplex serial ports. If the on-chip UART is being used as an inter-processor link, the PCA can be used to interface the 83C51FA to additional asynchronous lines.

This application uses several different Compare/Capture modes available on the PCA to receive or transmit bytes of data. It is assumed the reader is familiar the PCA and ASM51. For more information on the PCA refer to the "Hardware Description of the 83C51FA" chapter in the Embedded Controller Handbook (Order No. 210918).

Introduction

The figure below shows the format of a standard 10-bit asynchronous frame: 1 start bit (0), 8 data bits, and 1 stop bit (1). The start bit is used to synchronize the receiver to the transmitter; at the leading edge of the start bit the receiver must set up its timing logic to sample the incoming line in the center of each bit. Following the start bit are eight data bits which are transmitted least significant bit first. The stop bit is set to the opposite state of the start bit to guarantee that the leading edge of the start bit will cause a transition on the line. It also provides a dead time on the line so that the receiver can maintain its synchronization.

Two of the Compare/Capture modes on the PCA are used in receiving and transmitting data bits. When receiving, the Negative-Edge Capture mode allows the PCA to detect the start bit. Then using the Software Timer mode, interrupts are generated to sample the incoming data bits. This same mode is used to clock out bits when transmitting.

This Application Note contains four sections of code:

(1) List of variables
(2) Initialization routine
(3) Receive routine
(4) Transmit routine.

A complete listing of the routines and the test loop which was used to verify their operation is found in the Appendix. A total of three half-duplex channels were run at 2400 Baud in the test program. The listings shown here are simplified to one channel (Channel 0).

Variables

Listing 1 shows the variables used in both the receive and transmit routines. Flags are defined to signify the status of the reception or transmission of a byte (e.g. RCV_START_BIT, TXM_START_BIT). RCV_BUF and TXM_BUF simulate the on-chip serial port SBUF as two separate buffer registers. The temporary registers, RCV_REG and TXM_REG, are used to save bits as they are received or transmitted. Finally, two counter registers keep track of how many bits have been received or transmitted.

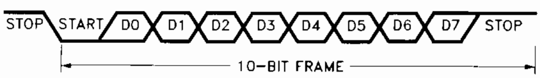
Variables are also needed to define one-half and one-full bit times in units of PCA timer ticks. (One bit time = 1 / baud rate.) With the PCA timer incremented every machine cycle, the equation to calculate one bit time can be written as:

$$\frac{\text{Osc. Freq.}}{(12)\cdot(\text{baud rate})} \quad \text{1 bit time (in PCA timer ticks)}$$

In this example, the baud rate is 2400 at 16 MHz.

$$\frac{16\ \text{MHz}}{(12)\cdot(2400)} = 556\ \text{counts} = \text{22C Hex}$$

The high and low byte of this value is placed in the variables FULL_BIT_HIGH and FULL_BIT_LOW, respectively. 115H is the value loaded into HALF_BIT_HIGH and HALF_BIT_LOW.



270531-1

Listing 1. Variables used by the software serial port. Channel 0

```
;
;  Receive Routine
;
RCV_START_BIT_0    BIT       20H.0      ; Indicates  start bit
                                        ; has been received
RCV_DONE_0         BIT       20H.1      ; Indicates  data byte
                                        ; has been received
RCV_BUF_0          DATA      30H        ; Software Receive
                                        ; "SBUF"
RCV_REG_0          DATA      31H        ; Temporary register
                                        ; for receive bits
RCV_COUNT_0        DATA      32H        ; Counter for receiving
                                        ; bits
;  Transmit Routine:
;
TXM_START_BIT_0    BIT       20H.3      ; Indicates start bit
                                        ; has been transmitted
TXM_IN_PROGRESS_0  BIT       20H.4      ; Indicates transmit is
                                        ; in progress
TXM_BUF_0          DATA      34H        ; Software transmit
                                        ; "SBUF"
TXM_REG_0          DATA      35H        ; Temporary register
                                        ; for transmitting bits
TXM_COUNT_0        DATA      36H        ; Counter for transmit-
                                        ; ting bits
DATA_0             DATA      37H        ; Register used for the
                                        ; test program
;
NEG_EDGE           EQU       11H        ; Two modes of operation
S_W_TIMER          EQU       49H        ; for compare/capture
                                        ; modules
;
HALF_BIT_HIGH      EQU       01H        ; Half bit time = 115H
HALF_BIT_LOW       EQU       15H
FULL_BIT_HIGH      EQU       02H        ; Full bit time = 22CH
FULL_BIT_LOW       EQU       2CH        ; 2400 Baud at 16 MHz
```

270531-4

Initialization

Listing 2 contains the intialization code for the receive and transmit process. Module 0 of the PCA is used as a receiver and is first set up to detect a negative edge from the start bit. Modules 2 and 3 are used for the additional 2 channels (see the Appendix). Module 3 is used as a separate software timer to transmit bits.

Listing 2. Initialization Routine

```
ORG 0000H
LJMP INITIALIZE
ORG 001BH
LJMP RECEIVE_DONE                    ; Timer 1 overflow -
                                     ; simulates "RI" interrupt
ORG 0033H
LJMP RECEIVE                         ; PCA interrupt
;
INITIALIZE: MOV SP, #5FH             ; Initialize stack pointer
                                     ; (specific to test program)
INIT_PCA: MOV CMOD, #00H             ; Increment PCA timer
                                     ; @ 1/12 Osc Frequency
          MOV CCON, #00H             ; Clear all status flags
          MOV CCAPM0, #NEG_EDGE      ; Module 0 in negative-edge
                                     ; trigger mode (P1.3)
          MOV CCAPM3, #S_W_TIMER     ; Module 3 as software timer
                                     ; mode
          MOV CL, #00H
          MOV CH, #00H
          MOV IE, #0D8H              ; Init all needed interrupts
                                     ; EA, EC, ES, ET1
          SETB CR                    ; Turn on PCA Counter
```

270531-5

All flags and registers from Listing 1 should be cleared in the initialization process.

Receive Routine

Two operating modes of the PCA are needed to receive bits. The module must first be able to detect the leading edge of a start bit so it is initially set up to capture a 1-to-0 transition (i.e. Negative-Edge Capture mode). The module is then reconfigured as a software timer to cause an interrupt at the center of each bit to deserialize the incoming data. The flowchart for the receive routine is given in Figure 1.

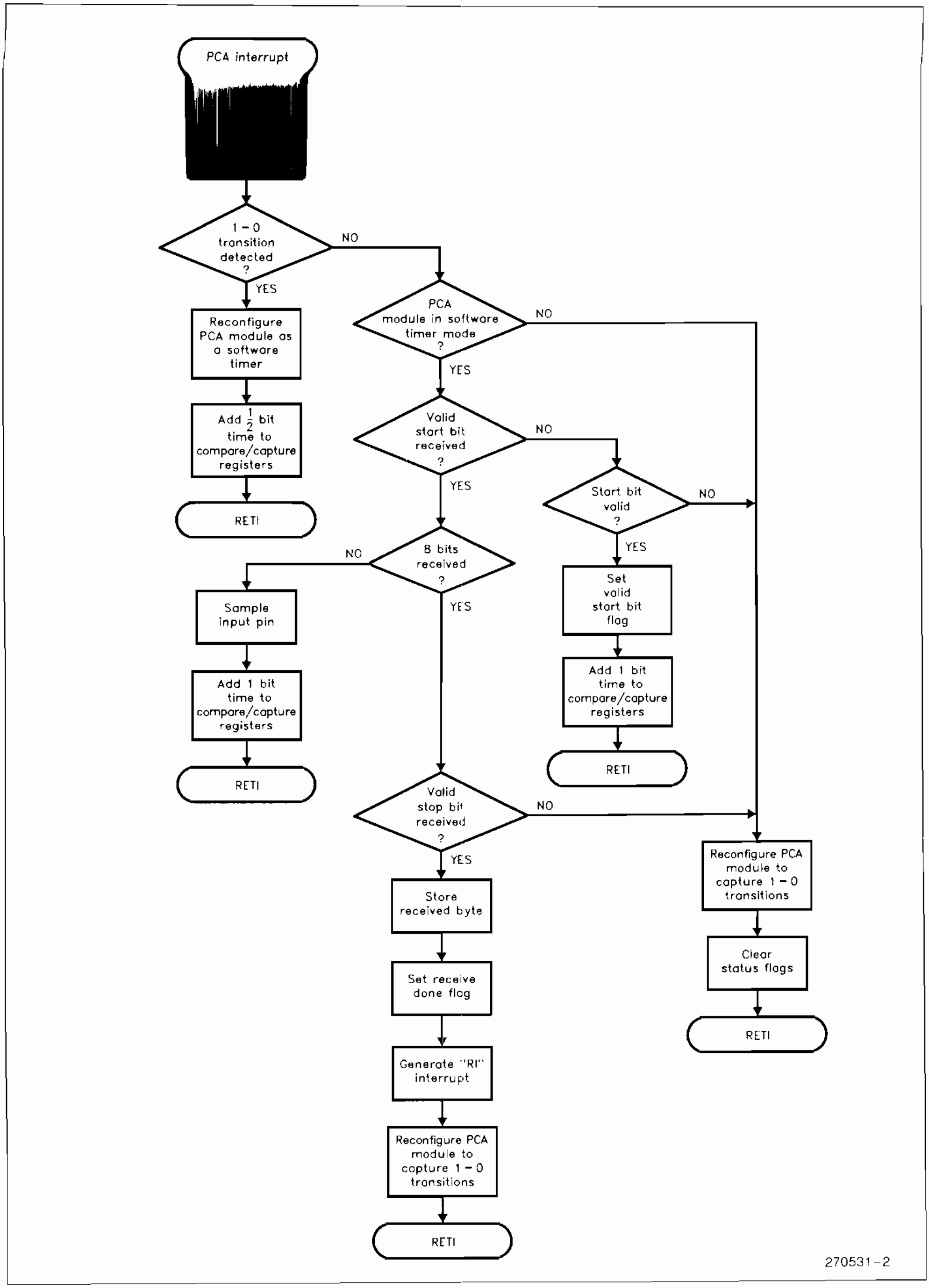


Figure 1. Flowchart for the Receive Routine

Listing 3.1 shows the code needed to detect a start bit. Notice that the first software timer interrupt will occur one-half bit time after the leading edge of the start bit to check its validity. If it is valid, the RCV_START_BIT is set. The rest of the samples will occur a full bit time later. The RCV_COUNT register is loaded with a value of 9 which indicates the number of bits to be sampled: 8 data bits and 1 stop bit.

Listing 3.1. Receive Interrupt Routine

```
RECEIVE:   PUSH ACC
           PUSH PSW
;
MODULE_0:  CLR CCF0                  ; Assume reception on
                                     ; Module 0
           MOV A, CCAPM0             ; Check mode of module.  If
           ANL A, #01111111B         ; set up to receive negative
           CJNE A,#NEG_EDGE, RCV_START_0 ; edges, then module
                                     ; is waiting for a start  bit
;
           CLR C                     ; Update compare/capture
           MOV A, #HALF_BIT_LOW      ; registers for half bit time
           ADD A, CCAP0L             ; to sample start bit
           MOV CCAP0L, A             ; Half bit time = 115H
           MOV A, #HALF_BIT_HIGH
           ADDC A, CCAP0H
           MOV CCAP0H, A
           MOV CCAPM0, #S_W_TIMER ; Reconfigure module 0 as
           POP PSW                   ; a software timer to sample
           POP ACC                   ; bits
           RETI
;
RCV_START_0: CJNE A, #S_W_TIMER, ERROR_0 ; Check module is
                                     ; configured as a software
                                     ; timer, otherwise error.
           JB RCV_START_BIT_0, RCV_BYTE_0 ; Check if start bit
                                     ; is received yet.
           JB P1.3, ERROR_0          ; Check that start bit = 0,
                                     ; otherwise error.
           SETB RCV_START_BIT_0      ; Signify valid start bit
                                     ; was received
           MOV RCV_COUNT_0, #09H     ; Start counting bits sampled
;
           CLR C                     ; Update compare/capture
           MOV A, #FULL_BIT_LOW      ; registers to sample
           ADD A, CCAP0L             ; incoming bits
           MOV CCAP0L, A             ; Full bit time = 22CH
           MOV A, #FULL_BIT_HIGH
           ADDC A, CCAP0H
           MOV CCAP0H, A
           POP PSW
           POP ACC
           RETI
```

270531-6

The next 8 timer interrupts will receive the incoming data bits; the RCV_COUNT register keeps track of how many bits have been sampled. As each bit is sampled, it is shifted through the Carry Flag and saved in RCV_REG. The ninth sample checks the validity of the stop bit. If it is valid, the data byte is moved into RCV BUF.

The main routine must have a way to know that a byte has been received. With the on-chip UART, the RI (Receive Interrupt) bit is set whenever a byte has been received. For the software serial port, any unimplemented interrupt vector can be used to generate an interrupt when a byte has been received. This routine uses the Timer 1 Overflow interrupt (its selection is arbitrary). A routine to test this interrupt is included in the listing in the Appendix.

Listing 3.2. Receive Interrupt Routine (Continued)

```
RCV_BYTE_0: DJNZ RCV_COUNT_0, RCV_DATA_0   ; On 9th sample,
                                          ; check for valid stop bit
RCV_STOP_0: JNB P1.3, ERROR_0
            MOV RCV_BUF_0, RCV_REG_0 ; Save received byte in
                                     ; receive "SBUF"
            SETB RCV_DONE_0          ; Flag which module received
                                     ; a byte
            SETB TF1                 ; Generate an interrupt so
                                     ; main program knows a byte
                                     ; has been received
                                     ; (Note: selection of TF1 is
                                     ; arbitrary)
            MOV CCAPM0, #NEG_EDGE    ; Reconfigure module 0 for
                                     ; Reception of a start bit
            POP PSW
            POP ACC
            RETI
;
RCV_DATA_0: MOV C, P1.3              ; Sampling data bits
            MOV A, RCV_REG_0         ; Shifts bits thru CY into
            RRC A                    ; ACC
            MOV RCV_REG_0, A         ; Save each reception in
                                     ; temporary register
            CLR C                    ; Update c/c register for
            MOV A, #FULL_BIT_LOW     ; next sample time
            ADD A, CCAP0L
            MOV CCAP0L, A
            MOV A, #FULL_BIT_HIGH
            ADDC A, CCAP0H
            MOV CCAP0H, A
            POP PSW
            POP ACC
            RETI
```

270531-7

In addition, an error routine (Listing 3.3) is included for invalid start or stop bits to offer some protection against noise. If an error occurs, the module is re-initialized to look for another start bit.

Listing 3.3 Error Routine for Receive Routine

```
ERROR_0:    MOV CCAPM0, #NEG_EDGE    ; Reset module to look for
                                     ; start bit
            CLR RCV_START_BIT_0      ; Clear flags which might
                                     ; have been set
            POP PSW
            POP ACC
            RETI
```

270531-8

Transmit Routine

Another PCA module is configured as a software timer to interrupt the CPU every bit time. With each timer interrupt one or more bits can be transmitted through port pins. In the test program three channels were operated simultaneously, but in the listings below, one channel is shown for simplicity. The selection of port pins is user programmable. The flowchart for the transmit routine is given in Figure 2.

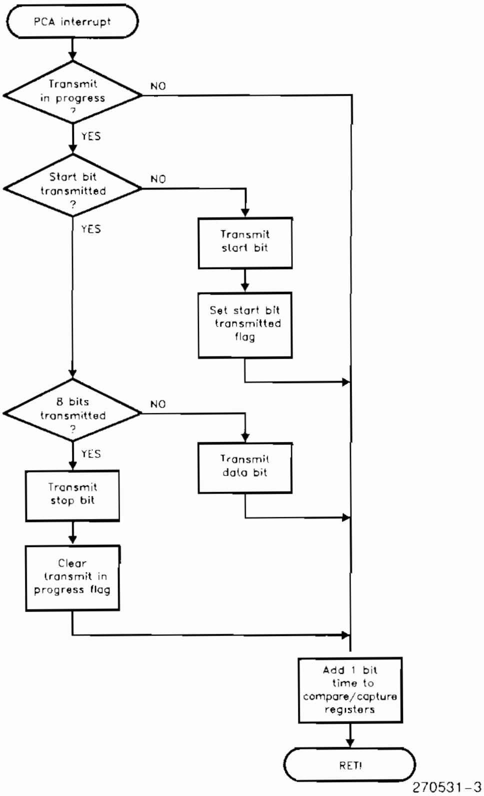


Figure 2. Flowchart for the Transmit Routine

When a byte is ready to be transmitted, the main program moves the data byte into the TXM_BUF register and sets the corresponding TXM_IN_PROGRESS bit. This bit informs the interrupt routine which channel is transmitting. The data byte is then moved in the storage register TXM_REG, and the TXM_COUNT is loaded. This main routine is shown in Listing 4.1.

Listing 4.1 Transmit Set Up Routine. Channel 0.

```
TXM_ON_0: CLR TXM_START_BIT_0     ; Clear status flag from
                                  ; previous transmission
          MOV TXM_BUF_0, DATA_0   ; Load "SBUF" with data byte
          MOV TXM_REG_0, TXM_BUF_0
          MOV TXM_COUNT_0, #09    ; 8 data bits + 1 stop bit
          SETB TXM_IN_PROGRESS_0
```

270531-9

Listing 4.2 shows the transmit interrupt routine. The first time through, the start bit is transmitted. As each successive interrupt outputs a bit, the contents of TXM_REG is shifted right one place into the Carry flag, and the TXM_COUNT is decremented. When TXM_COUNT equals zero, the stop bit is transmitted.

Listing 4.2. Transmit Interrupt Routine

```
TRANSMIT: PUSH ACC
          PUSH PSW
          CLR CCF3                ; Clear s/w timer interrupt
                                  ; for transmitting bits
          JNB TXM_IN_PROGRESS_0, TRANSMIT_1 ; Check which
                                  ; channel is transmitting.
                                  ; "TRANSMIT_1" is listed in
                                  ; the Appendix
;
TRANSMIT_0: JB TXM_START_BIT_0, TXM_BYTE_0  ; If start bit
                                  ; has been sent, continue
                                  ; transmitting bits.
          CLR P3.2                ; Otherwise transmit start
                                  ; bit
          SETB TXM_START_BIT_0    ; Signify start bit sent
          JMP TXM_EXIT
;
TXM_BYTE_0: DJNZ TXM_COUNT_0, TXM_DATA_0  ; If bit count
                                  ; equals 1 thru 9, transmit
                                  ; data bits (8 total)
;
TXM_STOP_0: SETB P3.2             ; When bit count = 0,
                                  ; transmit stop bit
          CLR TXM_IN_PROGRESS_0   ; Indicate transmission is
                                  ; finished and ready for
                                  ; next byte
          JMP TXM_EXIT
;
TXM_DATA_0: MOV A, TXM_REG_0      ; Transmit one bit at a time
          RRC A                   ; through the carry bit
          MOV P3.2, C
          MOV TXM_REG_0, A        ; Save what's not been sent
;
TXM_EXIT: CLR C                   ; Update compare value with
          MOV A, #FULL_BIT_LOW    ; Full bit time = 22CH
          ADD A, CCAP3L
          MOV CCAP3L, A
          MOV A, #FULL_BIT_HIGH
          ADDC A, CCAP3H
          MOV CCAP3H, A
          POP PSW
          POP ACC
          RETI
```

270531-10

Conclusion

The software routines in the Appendix can be altered to vary the baud rate and number of channels to fit a particular application. The number of channels which can be implemented is limited by the CPU time required to service the PCA interrupt. At higher baud rates, fewer channels can be run.

The test program verifies the simultaneous operation of three half-duplex channels at 2400 Baud and the on-chip full-duplex channel at 9600 Baud. Thirty-three percent of the CPU time is required to operate all four channels. The test was run for several hours with no apparent malfunctions.

APPENDIX

```
MCS-51 MACRO ASSEMBLER    SWPORT                                                    01/01/80    PAGE    1

DOS 3.20 (038-N) MCS-51 MACRO ASSEMBLER, V2.2
OBJECT MODULE PLACED IN SWPORT.OBJ
ASSEMBLER INVOKED BY:  C:\AEDIT\ASM51.EXE SWPORT.RCV

LOC  OBJ            LINE     SOURCE

                       1     $NOMOD51
                       2     $NOSYMBOLS
                       3     $NOLIST
                     152     ;
                     153     ;
                     154     ;
                     155     ; This program tests the receive routines of a software serial port.
                     156     ; Three half-duplex channels are implemented in software to run at
                     157     ; 2400 Baud (16MHz).  The on-chip serial port is also running full-duplex
                     158     ; at 9600 Baud.  Thirty-three percent of the CPU time is required to run
                     159     ; all four ports simultaneously.
                     160     ;
                     161     ; To test the receive routines, "dummy" terminals transmit 00 - FF hex
                     162     ; continually to the PCA.  When the first byte is received, it is
                     163     ; compared with 00.  If the comparison is valid, the compare value is
                     164     ; incremented and the routine waits to receive the next byte.  Error
                     165     ; routines toggle various Port 3 pins if an invalid comparison occurs
                     166     ; or if an invalid start bit or stop bit is received.
                     167     ;
0000                 168     ORG 00H
0000 020036          169     LJMP INITIALIZE
                     170     ;
001B                 171     ORG 001BH
001B 02025C          172     LJMP RECEIVE_DONE             ; Timer 1 Overflow - simulates "RI" interrupt
                     173     ;
0023                 174     ORG 0023H
0023 020282          175     LJMP SERIAL_PORT              ; Serial port interrupt
                     176     ;
0033                 177     ORG 0033H                     ; PCA interrupt
0033 0200DC          178     LJMP RECEIVE
                     179     ;
                     180     ;
                     181     ;                   VARIABLES USED BY THE SOFTWARE SERIAL PORT
                     182     ;                   ==========================================
                     183     ;
                     184     ;  RECEIVE ROUTINE:
                     185     ;  ---------------
                     186     ;
  0000               187     RCV_START_BIT_0      BIT            20H.0    ; Indicates start bit has been
  0008               188     RCV_START_BIT_1      BIT            21H.0    ; received
  0010               189     RCV_START_BIT_2      BIT            22H.0
                     190     ;
  0001               191     RCV_DONE_0           BIT            20H.1    ; Indicates data byte has been
  0009               192     RCV_DONE_1           BIT            21H.1    ; received
  0011               193     RCV_DONE_2           BIT            22H.1
                     194     ;
  0002               195     RCV_ON_0             BIT            20H.2    ; Used in main test program to check
  000A               196     RCV_ON_1             BIT            21H.2    ; for a received byte
  0012               197     RCV_ON_2             BIT            22H.2
                     198     ;
```

270531-11

```
MCS-51 MACRO ASSEMBLER    SWPORT                                        01/01/80   PAGE    2

LOC  OBJ          LINE    SOURCE

 0030                     RCV_BUF_0          DATA        30H     ; Software receive "SBUF"
 0040                     RCV_BUF_1          DATA        40H
 0050                     RCV_BUF_2          DATA        50H
                          ;
 0031                     RCV_REG_0          DATA        31H     ; Temporary register for
 0041                     RCV_REG_1          DATA        41H     ; receiving bits
 0051                     RCV_REG_2          DATA        51H
                          ;
 0032                     RCV_COUNT_0        DATA        32H     ; Counter for receiving bits
 0042                     RCV_COUNT_1        DATA        42H
 0052                     RCV_COUNT_2        DATA        52H
                          ;
 0033                     COUNT_0            DATA        33H     ; Used in test program to check
 0043                     COUNT_1            DATA        43H     ; bytes being received
 0053                     COUNT_2            DATA        53H
                          ;
 0011                     NEG_EDGE           EQU         11H     ; Two modes of operation for the
 0049                     S_W_TIMER          EQU         49H     ; Compare/Capture modules
                          ;
 0015                     HALF_BIT_LOW       EQU         15H     ; Half bit time = 115H
 0001                     HALF_BIT_HIGH      EQU         01H
 002C                     FULL_BIT_LOW       EQU         2CH     ; Full bit time = 22CH
 0002                     FULL_BIT_HIGH      EQU         02H
                                                                 ; 2400 Baud @ 16MHz
                          ;
                          ;

                          ;
                          ;                   INITIALIZATION ROUNTINE
                          ;                   ======================
                          ;
0036 75815F               INITIALIZE:    MOV SP, #5FH            ; Initialize stack pointer
                                                                 ; (specific to the test program)
0039 75D900               INIT_PCA:      MOV CMOD, #00H          ; Increment PCA clock @ 1/12 Osc Freq
003C 75D800                              MOV CCON, #00H          ; Clear all status flags
003F 75DA11                              MOV CCAPM0, #NEG_EDGE   ; Module 0 in Neg-edge capture mode (P1.3)
0042 75DB11                              MOV CCAPM1, #NEG_EDGE   ; Module 1        "                 (P1.4)
0045 75DC11                              MOV CCAPM2, #NEG_EDGE   ; Module 2        "                 (P1.5)
                          ;
0048 75E900                              MOV CL, #00H
004B 75F900                              MOV CH, #00H
004E 75A8D8                              MOV IE, #0D8H           ; Initialize needed interrupt: EA,EC,ES,ET1
0051 D2DE                                SETB CR                 ; Turn on PCA counter
                          ;
0053 759850               INIT_SP:       MOV SCON, #50H          ; Serial port in mode 1 (8-Bit UART)
0056 75CBFF                              MOV RCAP2H, #0FFH       ; Reload values for 9600 Baud @ 16 MHz
0059 75CACC                              MOV RCAP2L, #0CCH
005C 75C834                              MOV T2CON, #34H         ; Timer 2 as a baud-rate generator,
                                                                 ; turn on timer 2
                          ;
005F C200                 INIT_FLAGS:    CLR RCV_START_BIT_0
0061 C208                                CLR RCV_START_BIT_1
0063 C210                                CLR RCV_START_BIT_2
                          ;
0065 C201                                CLR RCV_DONE_0
```

270531-12

```
MCS-51 MACRO ASSEMBLER    SWPORT                                    01/01/80    PAGE    3

LOC  OBJ          SOURCE

0067 C209                        CLR RCV_DONE_1
0069 C211                        CLR RCV_DONE_2
                  ;
006B C202                        CLR RCV_ON_0
006D C20A                        CLR RCV_ON_1
006F C212                        CLR RCV_ON_2
                  ;
                  ; Port 3 pins used in test program for error routines
                  ;
                  ; Main program:
0071 D2B2                        SETB P3.2                  ; Error in comparison on module 0
0073 D2B3                        SETB P3.3                  ; Error in comparison on module 1
0075 D2B4                        SETB P3.4                  ; Error in comparison on module 2
                  ;
                  ; Interrupt routines:
0077 D2B5                        SETB P3.5                  ; Error in reception on module 0
0079 D2B6                        SETB P3.6                  ; Error in reception on module 1
007B D2B7                        SETB P3.7                  ; Error in reception on module 2
                  ;
007D 753000                      MOV RCV_BUF_0, #00H
0080 754000                      MOV RCV_BUF_1, #00H
0083 755000                      MOV RCV_BUF_2, #00H
                  ;
0086 753200                      MOV RCV_COUNT_0, #00H
0089 754200                      MOV RCV_COUNT_1, #00H
008C 755200                      MOV RCV_COUNT_2, #00H
                  ;
008F 753100                      MOV RCV_REG_0, #00H
0092 754100                      MOV RCV_REG_1, #00H
0095 755100                      MOV RCV_REG_2, #00H
                  ;
0098 753300                      MOV COUNT_0, #00H
009B 754300                      MOV COUNT_1, #00H
009E 755300                      MOV COUNT_2, #00H
                  ;
                  ;
                  ;
                  ;                       MAIN TEST ROUTINE - RECEIVE BITS
                  ;                       ================================
                  ;
00A1 300209       CHECK_0:       JNB RCV_ON_0, CHECK_1      ; Main program continually checks
00A4 E530                        MOV A, RCV_BUF_0           ; each channel for a received byte.
00A6 B5331E                      CJNE A, COUNT_0, ERROR0    ; Once a byte is received, it is compared
00A9 C202                        CLR RCV_ON_0               ; with the current value in the "COUNT"
00AB 0533                        INC COUNT_0                ; register
                  ;
00AD 300A09       CHECK_1:       JNB RCV_ON_1, CHECK_2
00B0 E540                        MOV A, RCV_BUF_1
00B2 B54319                      CJNE A, COUNT_1, ERROR1
00B5 C20A                        CLR RCV_ON_1
00B7 0543                        INC COUNT_1
                  ;
00B9 3012E5       CHECK_2:       JNB RCV_ON_2, CHECK_0
00BC E550                        MOV A, RCV_BUF_2
00BE B55314                      CJNE A, COUNT_2, ERROR2
```

270531-13

```
MCS-51 MACRO ASSEMBLER    SWPORT                                                  01/01/80   PAGE    4

LOC  OBJ            LINE     SOURCE

00C1 C212            309                  CLR RCV_ON_2
00C3 0553            310                  INC COUNT_2
00C5 80DA            311                  JMP CHECK_0
                     312     ;
00C7 C2B2            313     ERROR0:      CLR P3.2                          ; Error in comparison on module 0
00C9 75DA00          314                  MOV CCAPM0, #00H                  ; Discontinue receiving bytes
00CC 80DF            315                  JMP CHECK_1
                     316     ;
00CE C2B3            317     ERROR1:      CLR P3.3                          ; Error in comparison on module 1
00D0 75DB00          318                  MOV CCAPM1, #00H
00D3 80E4            319                  JMP CHECK_2
                     320     ;
00D5 C2B4            321     ERROR2:      CLR P3.4                          ; Error in comparison on module 2
00D7 75DC00          322                  MOV CCAPM2, #00H
00DA 80C5            323                  JMP CHECK_0
                     324     ;
                     325     ;
                     326     ;
                     327     ;                 PCA INTERRUPT ROUNTINE - RECEIVE BITS
                     328     ;                 ====================================
                     329     ;
00DC C0E0            330     RECEIVE:     PUSH ACC
00DE C0D0            331                  PUSH PSW
                     332     ;
00E0 20D811          333                  JB CCF0, MODULE_0                 ; Check which module caused
00E3 20D908          334                  JB CCF1, JUMP_1                   ; PCA interrupt and jump to
00E6 20DA08          335                  JB CCF2, JUMP_2                   ; appropriate routine
00E9 D0D0            336                  POP PSW
00EB D0E0            337                  POP ACC
00ED 32              338                  RETI
                     339     ;
00EE 02016C          340     JUMP_1:      LJMP MODULE_1
00F1 0201E4          341     JUMP_2:      LJMP MODULE_2
                     342     ;
                     343     ;
                     344     ;
                     345     ;                 CHANNEL 0
                     346     ;                 ---------
                     347     ;
00F4 C2D8            348     MODULE_0:    CLR CCF0                          ; Reception on module 0
00F6 E5DA            349                  MOV A, CCAPM0                     ; Check mode of module. If set up to
00F8 547F            350                  ANL A, #01111111B                 ; receive negative edges, then module
00FA B41115          351                  CJNE A, #NEG_EDGE, RCV_START_0    ; is waiting for a start bit
                     352     ;
00FD C3              353                  CLR C                             ; Update Compare/Capture registers for
00FE 7415            354                  MOV A, #HALF_BIT_LOW              ; half a bit time
0100 25EA            355                  ADD A, CCAP0L                     ; to sample start bit
0102 F5EA            356                  MOV CCAP0L, A                     ; Half bit time = 115H
0104 7401            357                  MOV A, #HALF_BIT_HIGH
0106 35FA            358                  ADDC A, CCAP0H
0108 F5FA            359                  MOV CCAP0H, A
010A 75DA49          360                  MOV CCAPM0, #S_W_TIMER            ; Reconfigure module 0 as
                     361                                                    ; a software timer to sample bits
010D D0D0            362                  POP PSW
010F D0E0            363                  POP ACC
```

270531-14

```
MCS-51 MACRO ASSEMBLER    SWPORT                                          01/01/80   PAGE    5

LOC  OBJ        SOURCE

0111 32                         RETI
                ;
0112 B4494B     RCV_START_0:    CJNE A, #S_W_TIMER, ERROR_0      ; Check module is configured
                                                                 ; as a software timer, otherwise error.
0115 20001A                     JB RCV_START_BIT_0, RCV_BYTE_0   ; Check if start bit
                                                                 ; has been received yet
0118 209345                     JB P1.3, ERROR_0                 ; Check that start bit = 0,
                                                                 ; otherwise error.
011B D200                       SETB RCV_START_BIT_0             ; Signify valid start bit
                                                                 ; was received
011D 753209                     MOV RCV_COUNT_0, #09H            ; Start counting bits sampled
                ;
0120 C3                         CLR C                            ; Update C/C registers to sample
0121 742C                       MOV A, #FULL_BIT_LOW             ; incoming bits
0123 25EA                       ADD A, CCAPOL                    ; Full bit time = 22CH
0125 F5EA                       MOV CCAPOL, A
0127 7402                       MOV A, #FULL_BIT_HIGH
0129 35FA                       ADDC A, CCAPOH
012B F5FA                       MOV CCAPOH, A
012D D0D0                       POP PSW
012F D0E0                       POP ACC
0131 32                         RETI
                ;
0132 D53212     RCV_BYTE_0:     DJNZ RCV_COUNT_0, RCV_DATA_0     ; On 9th sample, check for
                                                                 ; valid stop bit
0135 309328     RCV_STOP_0:     JNB P1.3, ERROR_0
0138 853130                     MOV RCV_BUF_0, RCV_REG_0         ; Save received byte in receive "SBUF"
013B D201                       SETB RCV_DONE_0                  ; Flag which module received a byte
013D D28F                       SETB TF1                         ; Generate an interrupt so main program
                                                                 ; knows a byte has been received
                                                                 ; (NOTE: selection of TF1 is arbitrary)
013F 75DA11                     MOV CCAPM0, #NEG_EDGE            ; Reconfigure module 0 for next
0142 D0D0                       POP PSW                          ; reception of a start bit
0144 D0E0                       POP ACC
0146 32                         RETI
                ;
0147 A293       RCV_DATA_0:     MOV C, P1.3                      ; Sampling data bits
0149 E531                       MOV A, RCV_REG_0                 ; Shift bits through CY into ACC
014B 13                         RRC A
014C F531                       MOV RCV_REG_0, A                 ; Save each reception in temporary
                                                                 ; register
014E C3                         CLR C
014F 742C                       MOV A, #FULL_BIT_LOW             ; Update C/C register for next
0151 25EA                       ADD A, CCAPOL                    ; sample time
0153 F5EA                       MOV CCAPOL, A
0155 7402                       MOV A, #FULL_BIT_HIGH
0157 35FA                       ADDC A, CCAPOH
0159 F5FA                       MOV CCAPOH, A
015B D0D0                       POP PSW
015D D0E0                       POP ACC
015F 32                         RETI
                ;
0160 C2B5       ERROR_0:        CLR P3.5                         ; Error routine for invalid start or
                                                                 ; stop bit or invalid mode comparison
```

270531-15

```
MCS-51 MACRO ASSEMBLER    SWPORT                                                01/01/80   PAGE    6

LOC  OBJ            LINE     SOURCE

                     418                                          ; Port pin used for debug only
0162 75DA11          419                MOV CCAPM0, #NEG_EDGE     ; Reset module to look for start bit
0165 C200            420                CLR RCV_START_BIT_0       ; Clear flags which might have been set
0167 D0D0            421                POP PSW
0169 D0E0            422                POP ACC
016B 32              423                RETI
                     424     ;
                     425     ;
                     426     ;                    CHANNEL 1
                     427     ;                    ---------
                     428     ;
016C C2D9            429     MODULE_1:  CLR CCF1                  ; Similar to module 0
016E E5DB            430                MOV A, CCAPM1
0170 547F            431                ANL A, #01111111B
0172 B41115          432                CJNE A, #NEG_EDGE, RCV_START_1
                     433     ;
0175 C3              434                CLR C
0176 7415            435                MOV A, #HALF_BIT_LOW
0178 25EB            436                ADD A, CCAP1L
017A F5EB            437                MOV CCAP1L, A
017C 7401            438                MOV A, #HALF_BIT_HIGH
017E 35FB            439                ADDC A, CCAP1H
0180 F5FB            440                MOV CCAP1H, A
0182 75DB49          441                MOV CCAPM1, #S_W_TIMER
0185 D0D0            442                POP PSW
0187 D0E0            443                POP ACC
0189 32              444                RETI
                     445     ;
018A B4494B          446     RCV_START_1:  CJNE A, #S_W_TIMER, ERROR_1
018D 20081A          447                JB RCV_START_BIT_1, RCV_BYTE_1
0190 209445          448                JB P1.4, ERROR_1
                     449     ;
0193 D208            450                SETB RCV_START_BIT_1
0195 754209          451                MOV RCV_COUNT_1, #09H
                     452     ;
0198 C3              453                CLR C
0199 742C            454                MOV A, #FULL_BIT_LOW
019B 25EB            455                ADD A, CCAP1L
019D F5EB            456                MOV CCAP1L, A
019F 7402            457                MOV A, #FULL_BIT_HIGH
01A1 35FB            458                ADDC A, CCAP1H
01A3 F5FB            459                MOV CCAP1H, A
01A5 D0D0            460                POP PSW
01A7 D0E0            461                POP ACC
01A9 32              462                RETI
                     463     ;
01AA D54212          464     RCV_BYTE_1:   DJNZ RCV_COUNT_1, RCV_DATA_1
                     465     ;
01AD 309428          466     RCV_STOP_1:   JNB P1.4, ERROR_1
01B0 854140          467                MOV RCV_BUF_1, RCV_REG_1
01B3 D209            468                SETB RCV_DONE_1
01B5 D28F            469                SETB TF1
01B7 75DB11          470                MOV CCAPM1, #NEG_EDGE
01BA D0D0            471                POP PSW
01BC D0E0            472                POP ACC
```

270531-16

```
MCS-51 MACRO ASSEMBLER    SWPORT                                           01/01/80   PAGE    7

LOC  OBJ            LINE     SOURCE

01BE 32              473                     RETI
                     474     ;
01BF A294            475     RCV_DATA_1:     MOV C, P1.4
01C1 E541            476                     MOV A, RCV_REG_1
01C3 13              477                     RRC A
01C4 F541            478                     MOV RCV_REG_1, A
                     479     ;
01C6 C3              480                     CLR C
01C7 742C            481                     MOV A, #FULL_BIT_LOW
01C9 25EB            482                     ADD A, CCAP1L
01CB F5EB            483                     MOV CCAP1L, A
01CD 7402            484                     MOV A, #FULL_BIT_HIGH
01CF 35FB            485                     ADDC A, CCAP1H
01D1 F5FB            486                     MOV CCAP1H, A
01D3 D0D0            487                     POP PSW
01D5 D0E0            488                     POP ACC
01D7 32              489                     RETI
                     490     ;
01D8 C2B6            491     ERROR_1:        CLR P3.6
01DA 75DB11          492                     MOV CCAPM1, #NEG_EDGE
01DD C208            493                     CLR RCV_START_BIT_1
01DF D0D0            494                     POP PSW
01E1 D0E0            495                     POP ACC
01E3 32              496                     RETI
                     497     ;
                     498     ;
                     499     ;
                     500     ;                                CHANNEL 2
                     501     ;                                ---------
                     502     ;
                     503     ;
01E4 C2DA            504     MODULE_2:       CLR CCF2                                ; Similar to module 0
01E6 E5DC            505                     MOV A, CCAPM2
01E8 547F            506                     ANL A, #01111111B
01EA B41115          507                     CJNE A, #NEG_EDGE, RCV_START_2
                     508     ;
01ED C3              509                     CLR C
01EE 7415            510                     MOV A, #HALF_BIT_LOW
01F0 25EC            511                     ADD A, CCAP2L
01F2 F5EC            512                     MOV CCAP2L, A
01F4 7401            513                     MOV A, #HALF_BIT_HIGH
01F6 35FC            514                     ADDC A, CCAP2H
01F8 F5FC            515                     MOV CCAP2H, A
01FA 75DC49          516                     MOV CCAPM2, #S_W_TIMER
01FD D0D0            517                     POP PSW
01FF D0E0            518                     POP ACC
0201 32              519                     RETI
                     520     ;
0202 B4494B          521     RCV_START_2:    CJNE A, #S_W_TIMER, ERROR_2
0205 20101A          522                     JB RCV_START_BIT_2, RCV_BYTE_2
0208 209545          523                     JB P1.5, ERROR_2
                     524     ;
020B D210            525                     SETB RCV_START_BIT_2
020D 755209          526                     MOV RCV_COUNT_2, #09H
                     527     ;
```

270531-17

```
MCS-51 MACRO ASSEMBLER    SWPORT                                          01/01/80   PAGE    8

LOC  OBJ            LINE     SOURCE

0210 C3              528                     CLR C
0211 742C            529                     MOV A, #FULL_BIT_LOW
0213 25EC            530                     ADD A, CCAP2L
0215 F5EC            531                     MOV CCAP2L, A
0217 7402            532                     MOV A, #FULL_BIT_HIGH
0219 35FC            533                     ADDC A, CCAP2H
021B F5FC            534                     MOV CCAP2H, A
021D D0D0            535                     POP PSW
021F D0E0            536                     POP ACC
0221 32              537                     RETI
                     538     ;
0222 D55212          539     RCV_BYTE_2:     DJNZ RCV_COUNT_2, RCV_DATA_2
                     540     ;
0225 309528          541     RCV_STOP_2:     JNB P1.5, ERROR_2
0228 855150          542                     MOV RCV_BUF_2, RCV_REG_2
022B D211            543                     SETB RCV_DONE_2
022D D28F            544                     SETB TF1
022F 75DC11          545                     MOV CCAPM2, #NEG_EDGE
0232 D0D0            546                     POP PSW
0234 D0E0            547                     POP ACC
0236 32              548                     RETI
                     549     ;
0237 A295            550     RCV_DATA_2:     MOV C, P1.5
0239 E551            551                     MOV A, RCV_REG_2
023B 13              552                     RRC A
023C F551            553                     MOV RCV_REG_2, A
023E C3              554                     CLR C
023F 742C            555                     MOV A, #FULL_BIT_LOW
0241 25EC            556                     ADD A, CCAP2L
0243 F5EC            557                     MOV CCAP2L, A
0245 7402            558                     MOV A, #FULL_BIT_HIGH
0247 35FC            559                     ADDC A, CCAP2H
0249 F5FC            560                     MOV CCAP2H, A
024B D0D0            561                     POP PSW
024D D0E0            562                     POP ACC
024F 32              563                     RETI
                     564     ;
0250 C2B7            565     ERROR_2:        CLR P3.7
0252 75DC11          566                     MOV CCAPM2, #NEG_EDGE
0255 C210            567                     CLR RCV_START_BIT_2
0257 D0D0            568                     POP PSW
0259 D0E0            569                     POP ACC
025B 32              570                     RETI
                     571     ;
                     572     ;
                     573     ;
                     574     ;
                     575     ; This routine simulates the "RI" interrupt.  When a byte is received on one
                     576     ; of the channels, this interrupt is generated.  Bits are set so the main
                     577     ; routine knows which channel received a byte.
                     578     ;
                     579     ;
025C C0E0            580     RECEIVE_DONE:   PUSH ACC
025E C0D0            581                     PUSH PSW
0260 C28F            582                     CLR TF1
```

270531-18

```
MCS-51 MACRO ASSEMBLER    SWPORT                                                01/01/80   PAGE   9

LOC  OBJ            LINE     SOURCE

0262 300106          583                    JNB RCV_DONE_0, RCV_1     ; Check which module received a byte
0265 C201            584                    CLR RCV_DONE_0            ; Clear flags needed for next reception
0267 C200            585                    CLR RCV_START_BIT_0
0269 D202            586                    SETB RCV_ON_0             ; Tell main routine which channel received
                     587                                              ; a byte
026B 300906          588     RCV_1:         JNB RCV_DONE_1, RCV_2
026E C209            589                    CLR RCV_DONE_1
0270 C208            590                    CLR RCV_START_BIT_1
0272 D20A            591                    SETB RCV_ON_1
                     592     ;
0274 301106          593     RCV_2:         JNB RCV_DONE_2, RETURN
0277 C211            594                    CLR RCV_DONE_2
0279 C210            595                    CLR RCV_START_BIT_2
027B D212            596                    SETB RCV_ON_2
                     597     ;
027D D0D0            598     RETURN:        POP PSW
027F D0E0            599                    POP ACC
0281 32              600                    RETI
                     601     ;
                     602     ;
                     603     ;
                     604     ;
                     605     ;              SERIAL PORT INTERRUPT
                     606     ;              =====================
                     607     ;
                     608     ; When a byte is received on the full-duplex serial port, it is then
                     609     ; transmitted back to a "dummy" terminal.  This terminal checks that the
                     610     ; byte it transmitted to the PCA is the same value it receives back.
                     611     ;
                     612     ;
0282 C0E0            613     SERIAL_PORT:   PUSH ACC
0284 C0D0            614                    PUSH PSW
0286 30980B          615                    JNB RI, TXM               ; Check whether RI or TI
0289 E599            616                    MOV A, SBUF               ; caused the interrupt
028B C298            617                    CLR RI
028D F599            618                    MOV SBUF, A
028F D0D0            619                    POP PSW
0291 D0E0            620                    POP ACC
0293 32              621                    RETI
                     622     ;
0294 C299            623     TXM:           CLR TI
0296 D0D0            624                    POP PSW
0298 D0E0            625                    POP ACC
029A 32              626                    RETI
                     627     ;
                     628     END

REGISTER BANK(S) USED: 0

ASSEMBLY COMPLETE, NO ERRORS FOUND
```

270531-19

```
MCS-51 MACRO ASSEMBLER    SWPORT                                                    01/01/80   PAGE    1


DOS 3.20 (038-N) MCS-51 MACRO ASSEMBLER, V2.2
OBJECT MODULE PLACED IN SWPORT.OBJ
ASSEMBLER INVOKED BY:  C:\AEDIT\ASM51.EXE SWPORT.TR


LOC  OBJ            LINE     SOURCE

                       1     $NOMOD51
                       2     $NOSYMBOLS
                       3     $NOLIST
                     152     ;
                     153     ;
                     154     ; This program tests the transmit routines for the software serial port.
                     155     ; To initialize the first transmission, the compare values are loaded before
                     156     ; the PCA timer is started.  Successive interrupts are generated every bit
                     157     ; time by the software timer.
                     158     ;
                     159     ; For test purposes, the data transmitted increments from 00 to FF hex.
                     160     ; "Dummy" terminals receive these bytes and display the bytes as they
                     161     ; are incremented.
                     162     ;
                     163     ;
0000                 164     ORG 00H
0000 020036          165     LJMP INIT_TXM
                     166     ;
0023                 167     ORG 0023H
0023 02014B          168     LJMP SERIAL_PORT                          ; Serial port interrupt
                     169     ;
0033                 170     ORG 0033H
0033 0200D0          171     LJMP TRANSMIT                             ; PCA software timer interrupt
                     172     ;
                     173     ;
                     174     ;                   VARIABLES USED BY THE SOFTWARE SERIAL PORT
                     175     ;                   ==========================================
                     176     ;
  0003               177     TXM_START_BIT_0        BIT           20H.3    ; Indicates start bit has been
  000B               178     TXM_START_BIT_1        BIT           21H.3    ; transmitted
  0013               179     TXM_START_BIT_2        BIT           22H.3
                     180     ;
  0004               181     TXM_IN_PROGRESS_0      BIT           20H.4    ; Indicates transmit is in progress
  000C               182     TXM_IN_PROGRESS_1      BIT           21H.4
  0014               183     TXM_IN_PROGRESS_2      BIT           22H.4
                     184     ;
  0034               185     TXM_BUF_0              DATA          34H      ; Software transmit "SBUF"
  0044               186     TXM_BUF_1              DATA          44H
  0054               187     TXM_BUF_2              DATA          54H
                     188     ;
  0035               189     TXM_REG_0              DATA          35H      ; Temporary register for
  0045               190     TXM_REG_1              DATA          45H      ; transmitting bits
  0055               191     TXM_REG_2              DATA          55H
                     192     ;
  0036               193     TXM_COUNT_0            DATA          36H      ; Counter for transmitting bits
  0046               194     TXM_COUNT_1            DATA          46H
  0056               195     TXM_COUNT_2            DATA          56H
                     196     ;
  0037               197     DATA_0                 DATA          37H      ; Register used for the test
  0047               198     DATA_1                 DATA          47H      ; program
```

270531-20

```
MCS-51 MACRO ASSEMBLER    SWPORT                                              01/01/80   PAGE    2

LOC  OBJ            LINE     SOURCE

  0057               199     DATA_2              DATA          57H
                     200     ;
  0049               201     S_W_TIMER           EQU           49H      ; Software timer mode for the
                     202                                                ; compare/capture module
  002C               203     FULL_BIT_LOW        EQU           2CH      ; Full bit time = 22CH
  0002               204     FULL_BIT_HIGH       EQU           02H      ; 2400 Baud at 16 MHz
                     205     ;
                     206     ;
                     207     ;                   INITIALIZATION
                     208     ;                   ==============
                     209     ;
0036 75815F          210     INIT_TXM:      MOV SP, #5FH                    ; (Compatible with receive routines)
                     211     ;
0039 75D900          212                    MOV CMOD, #00H                  ; Increment PCA timer @ 1/12 osc. freq.
003C 75D800          213                    MOV CCON, #00H                  ; Clear all status flags
003F 75F900          214                    MOV CH, #00H
0042 75E900          215                    MOV CL, #00H
0045 75DD49          216                    MOV CCAPM3, #S_W_TIMER          ; Module 3 configured as software timer
                     217     ;
0048 75A8D8          218                    MOV IE, #0D8H                   ; Initialize all needed interrupts
                     219
004B 759850          220     INIT_SP:       MOV SCON, #50H                  ; Serial port in mode 1 (8-bit UART)
004E 75CBFF          221                    MOV RCAP2H, #0FFH               ; Reload values for 9600 Baud @ 16 MHz
0051 75CACC          222                    MOV RCAP2L, #0CCH
0054 75C834          223                    MOV T2CON, #34H                 ; Timer 2 as a baud-rate generator,
                     224                                                    ; turn Timer 2 on
                     225     ;
0057 C203            226     INIT_FLAGS:    CLR TXM_START_BIT_0
0059 C20B            227                    CLR TXM_START_BIT_1
005B C213            228                    CLR TXM_START_BIT_2
                     229     ;
005D C204            230                    CLR TXM_IN_PROGRESS_0
005F C20C            231                    CLR TXM_IN_PROGRESS_1
0061 C214            232                    CLR TXM_IN_PROGRESS_2
                     233     ;
0063 753400          234                    MOV TXM_BUF_0, #00H
0066 754400          235                    MOV TXM_BUF_1, #00H
0069 755400          236                    MOV TXM_BUF_2, #00H
                     237     ;
006C 753500          238                    MOV TXM_REG_0, #00H
006F 754500          239                    MOV TXM_REG_1, #00H
0072 755500          240                    MOV TXM_REG_2, #00H
                     241     ;
0075 753600          242                    MOV TXM_COUNT_0, #00H
0078 754600          243                    MOV TXM_COUNT_1, #00H
007B 755600          244                    MOV TXM_COUNT_2, #00H
                     245     ;
007E 7537FF          246                    MOV DATA_0, #0FFH
0081 7547FF          247                    MOV DATA_1, #0FFH
0084 7557FF          248                    MOV DATA_2, #0FFH
                     249     ;
0087 75ED2C          250                    MOV CCAP3L, #2CH                ; Cause the first software timer to
008A 75FD02          251                    MOV CCAP3H, #02H                ; interrupt one bit time after
008D D2DE            252                    SETB CR                         ; PCA timer is started
                     253     ;
```

270531-21

```
MCS-51 MACRO ASSEMBLER    SWPORT                                    01/01/80   PAGE    3

LOC  OBJ        SOURCE

                ;
                ;              MAIN TEST ROUTINE - TRANSMIT BITS
                ;              ==================================
                ;
008F 02009D     FIRST_TXM:     JMP TXM_ON_0
                ;
0092 300408     MAIN_TXM:      JNB TXM_IN_PROGRESS_0, TXM_ON_0 ; Determine if ready to send
0095 300C16                    JNB TXM_IN_PROGRESS_1, TXM_ON_1 ; next byte. (i.e. transmit
0098 301424                    JNB TXM_IN_PROGRESS_2, TXM_ON_2 ; "not" in progress)
009B 80F5                      JMP MAIN_TXM                    ; Waiting for "TI" flag
                ;
                ;
009D C203       TXM_ON_0:      CLR TXM_START_BIT_0             ; Clear flag from previous
009F 0537                      INC DATA_0                      ; transmission
00A1 853734                    MOV TXM_BUF_0, DATA_0           ; Load "SBUF" with data byte
00A4 853435                    MOV TXM_REG_0, TXM_BUF_0
00A7 753609                    MOV TXM_COUNT_0, #09H           ; 8 data bits + 1 stop bit
00AA D204                      SETB TXM_IN_PROGRESS_0
00AC 80E4                      JMP MAIN_TXM
                ;
00AE C20B       TXM_ON_1:      CLR TXM_START_BIT_1
00B0 0547                      INC DATA_1
00B2 854744                    MOV TXM_BUF_1, DATA_1
00B5 854445                    MOV TXM_REG_1, TXM_BUF_1
00B8 754609                    MOV TXM_COUNT_1, #09H
00BB D20C                      SETB TXM_IN_PROGRESS_1
00BD 80D3                      JMP MAIN_TXM
                ;
00BF C213       TXM_ON_2:      CLR TXM_START_BIT_2
00C1 0557                      INC DATA_2
00C3 855754                    MOV TXM_BUF_2, DATA_2
00C6 855455                    MOV TXM_REG_2, TXM_BUF_2
00C9 755609                    MOV TXM_COUNT_2, #09H
00CC D214                      SETB TXM_IN_PROGRESS_2
00CE 80C2                      JMP MAIN_TXM
                ;
                ;
                ;              PCA INTERRUPT ROUTINE - TRANSMIT BITS
                ;              =====================================
                ;
00D0 C0E0       TRANSMIT:      PUSH ACC
00D2 C0D0                      PUSH PSW
00D4 C2DB                      CLR CCF3                        ; Clear s/w timer interrupt
00D6 30041E                    JNB TXM_IN_PROGRESS_0,TRANSMIT_1 ; Check which channel is
                                                               ; transmitting
                ;
                ;                        CHANNEL 0
                ;                        ---------
                
00D9 200307     TRANSMIT_0:    JB TXM_START_BIT_0, TXM_BYTE_0  ; If start bit has been sent,
                                                               ; continue transmitting data bits,
00DC C2B2                      CLR P3.2                        ; otherwise transmit start bit
00DE D203                      SETB TXM_START_BIT_0            ; Signify start bit sent
00E0 0200F7                    JMP TRANSMIT_1                  ; Check next transmit pin
                ;
```

270531-22

```
MCS-51 MACRO ASSEMBLER    SWPORT                                                          01/01/80   PAGE    4

LOC  OBJ            LINE     SOURCE

00E3 D53607          309     TXM_BYTE_0:     DJNZ TXM_COUNT_0, TXM_DATA_0     ; If bit count equals 1 thru 9,
                     310                                                      ; Transmit data bits (8 total)
00E6 D2B2            311     TXM_STOP_0:     SETB P3.2                        ; When bit count = 0, transmit stop bit
00E8 C204            312                     CLR TXM_IN_PROGRESS_0            ; Indicate transmission is finished and
                     313                                                      ; ready for next byte
00EA 0200F7          314                     JMP TRANSMIT_1                   ; Check next transmit pin
                     315     ;
00ED E535            316     TXM_DATA_0:     MOV A, TXM_REG_0                 ; Transmit one bit at a time
00EF 13              317                     RRC A                            ; through the carry bit
00F0 92B2            318                     MOV P3.2, C
00F2 F535            319                     MOV TXM_REG_0, A                 ; Save what's not been sent
00F4 0200F7          320                     JMP TRANSMIT_1                   ; Check next transmit pin
                     321     ;
                     322     ;                                   CHANNEL 1
                     323     ;                                   ---------
                     324     ;
00F7 300C1E          325     TRANSMIT_1:     JNB TXM_IN_PROGRESS_1, TRANSMIT_2 ; Similar to TRANSMIT_0
00FA 200B07          326                     JB TXM_START_BIT_1, TXM_BYTE_1
00FD C2B3            327                     CLR P3.3
00FF D20B            328                     SETB TXM_START_BIT_1
0101 020118          329                     JMP TRANSMIT_2
                     330     ;
0104 D54607          331     TXM_BYTE_1:     DJNZ TXM_COUNT_1, TXM_DATA_1
                     332     ;
0107 D2B3            333     TXM_STOP_1:     SETB P3.3
0109 C20C            334                     CLR TXM_IN_PROGRESS_1
010B 020118          335                     JMP TRANSMIT_2
                     336     ;
010E E545            337     TXM_DATA_1:     MOV A, TXM_REG_1
0110 13              338                     RRC A
0111 92B3            339                     MOV P3.3, C
0113 F545            340                     MOV TXM_REG_1, A
0115 020118          341                     JMP TRANSMIT_2
                     342     ;
                     343     ;                                   CHANNEL 2
                     344     ;                                   ---------
                     345     ;
0118 30141E          346     TRANSMIT_2:     JNB TXM_IN_PROGRESS_2, TXM_EXIT ; Similar to TRANSMIT_0
011B 201307          347                     JB TXM_START_BIT_2, TXM_BYTE_2
011E C2B4            348                     CLR P3.4
0120 D213            349                     SETB TXM_START_BIT_2
0122 020139          350                     JMP TXM_EXIT
                     351     ;
0125 D55607          352     TXM_BYTE_2:     DJNZ TXM_COUNT_2, TXM_DATA_2
                     353     ;
0128 D2B4            354     TXM_STOP_2:     SETB P3.4
012A C214            355                     CLR TXM_IN_PROGRESS_2
012C 020139          356                     JMP TXM_EXIT
                     357     ;
012F E555            358     TXM_DATA_2:     MOV A, TXM_REG_2
0131 13              359                     RRC A
0132 92B4            360                     MOV P3.4, C
0134 F555            361                     MOV TXM_REG_2, A
0136 020139          362                     JMP TXM_EXIT
                     363     ;
```

270531-23

```
MCS-51 MACRO ASSEMBLER    SWPORT                                                     01/01/80   PAGE    5

LOC  OBJ            LINE     SOURCE

0139 C3              364     TXM EXIT:     CLR C                          ; Update compare value with
013A 742C            365                   MOV A, #FULL BIT LOW           ; full bit time - 22CH
013C 25ED            366                   ADD A, CCAP3L
013E F5ED            367                   MOV CCAP3L, A
0140 7402            368                   MOV A, #FULL BIT HIGH
0142 35FD            369                   ADDC A, CCAP3H
0144 F5FD            370                   MOV CCAP3H, A
0146 D0D0            371                   POP PSW
0148 D0E0            372                   POP ACC
014A 32              373                   RETI
                     374     ;
                     375     ;
                     376     ;             SERIAL PORT INTERRUPT
                     377     ;             =====================
                     378     ;
                     379     ; When a byte is received on the full-duplex serial port, it is then
                     380     ; transmitted back to a "dummy" terminal.  This terminal checks that
                     381     ; the byte it transmitted to the PCA is the same value it receives back.
                     392     ;
                     383     ;
014B C0E0            384     SERIAL PORT:  PUSH ACC
014D C0D0            385                   PUSH PSW
014F 30980B          386                   JNB RI, TXM                    ; Check whether RI or TI
0152 E599            387                   MOV A, SBUF                    ; caused the interrupt
0154 C298            388                   CLR RI
0156 F599            389                   MOV SBUF, A
0158 D0D0            390                   POP PSW
015A D0E0            391                   POP ACC
015C 32              392                   RETI
                     393     ;
015D C299            394     TXM:          CLR TI
015F D0D0            395                   POP PSW
0161 D0E0            396                   POP ACC
0163 32              397                   RETI
                     398     ;
                     399     ;
                     400     END

REGISTER BANK(S) USED: 0

ASSEMBLY COMPLETE, NO ERRORS FOUND
```

270531-24

DOMESTIC SALES OFFICES

ALABAMA

†Intel Corp
5015 Bradford Dr., #2
Huntsville 35805
Tel: (205) 830-4010
FAX: (205) 837-2640

ARIZONA

†Intel Corp
11225 N. 28th Dr
Suite D-214
Phoenix 85029
Tel: (602) 869-4980
FAX: (602) 869-4294

Intel Corp
7225 N. Mona Lisa Rd
Suite 215
Tucson 85741
Tel: (602) 544-0227
FAX: (602) 544-0232

CALIFORNIA

†Intel Corp
21515 Vanowen Street
Suite 116
Canoga Park 91303
Tel: (818) 704-8500
FAX: (818) 340-1144

†Intel Corp
2250 E. Imperial Highway
Suite 218
El Segundo 90245
Tel: (213) 640-6040
FAX: (213) 640-7133

Intel Corp
1 Sierra Gate Plaza
Suite 280C
Roseville 95678
Tel: (916) 782-8086
FAX: (916) 782-8153

†Intel Corp
9665 Chesapeake Dr
Suite 325
San Diego 92123
Tel: (619) 292-8086
FAX: (619) 292-0628

†Intel Corp.*
400 N. Tustin Avenue
Suite 450
Santa Ana 92705
Tel: (714) 835-9642
TWX: 910-595-1114
FAX: (714) 541-9157

†Intel Corp.*
San Tomas 4
2700 San Tomas Expressway
2nd Floor
Santa Clara 95051
Tel: (408) 986-8086
TWX: 910-338-0255
FAX: (408) 727-2620

COLORADO

Intel Corp
4445 Northpark Drive
Suite 100
Colorado Springs 80907
Tel: (719) 594-6622
FAX: (303) 594-0720

†Intel Corp.*
650 S. Cherry St
Suite 915
Denver 80222
Tel: (303) 321-8086
TWX: 910-931-2289
FAX: (303) 322-8670

CONNECTICUT

†Intel Corp
301 Lee Farm Corporate Park
83 Wooster Heights Rd
Danbury 06810
Tel: (203) 748-3130
FAX: (203) 794-0339

FLORIDA

†Intel Corp
6363 N.W. 6th Way
Suite 100
Ft. Lauderdale 33309
Tel: (305) 771-0600
TWX: 510-956-9407
FAX: (305) 772-8193

†Intel Corp
5850 T.G. Lee Blvd
Suite 340
Orlando 32822
Tel: (407) 240-8000
FAX: (407) 240-8097

Intel Corp
11300 4th Street North
Suite 170
St. Petersburg 33716
Tel: (813) 577-2413
FAX: (813) 578-1607

GEORGIA

Intel Corp
20 Technology Parkway, N.W.
Suite 150
Norcross 30092
Tel: (404) 449-0541
FAX: (404) 605-9762

ILLINOIS

†Intel Corp.*
300 N. Martingale Road
Suite 400
Schaumburg 60173
Tel: (708) 605-8031
FAX: (708) 706-9762

INDIANA

†Intel Corp
8777 Purdue Road
Suite 125
Indianapolis 46268
Tel: (317) 875-0623
FAX: (317) 875-8938

IOWA

Intel Corp
1930 St. Andrews Drive N.E.
2nd Floor
Cedar Rapids 52402
Tel: (319) 393-1294

KANSAS

†Intel Corp
10985 Cody St
Suite 140, Bldg. D
Overland Park 66210
Tel: (913) 345-2727
FAX: (913) 345-2076

MARYLAND

†Intel Corp.*
10010 Junction Dr
Suite 200
Annapolis Junction 20701
Tel: (301) 206-2860
FAX: (301) 206-3677
(301) 206-3678

MASSACHUSETTS

†Intel Corp.*
Westford Corp. Center
3 Carlisle Road
2nd Floor
Westford 01886
Tel: (508) 692-3222
TWX: 710-343-6333
FAX: (508) 692-7867

MICHIGAN

†Intel Corp
7071 Orchard Lake Road
Suite 100
West Bloomfield 48322
Tel: (313) 851-8096
FAX: (313) 851-8770

MINNESOTA

†Intel Corp
3500 W. 80th St
Suite 360
Bloomington 55431
Tel: (612) 835-6722
TWX: 910-576-2867
FAX: (612) 831-6497

MISSOURI

†Intel Corp
4203 Earth City Expressway
Suite 131
Earth City 63045
Tel: (314) 291-1990
FAX: (314) 291-4341

NEW JERSEY

†Intel Corp.*
Parkway 109 Office Center
328 Newman Springs Road
Red Bank 07701
Tel: (201) 747-2233
FAX: (201) 747-0983

†Intel Corp
280 Corporate Center
75 Livingston Avenue
First Floor
Roseland 07068
Tel: (201) 740-0111
FAX: (201) 740-0626

NEW YORK

Intel Corp.*
850 Cross Keys Office Park
Fairport 14450
Tel: (716) 425-2750
TWX: 510-253-7391
FAX: (716) 223-2561

†Intel Corp.*
2950 Expressway Dr., South
Suite 130
Islandia 11722
Tel: (516) 231-3300
TWX: 510-227-6236
FAX: (516) 348-7939

†Intel Corp
Westage Business Center
Bldg. 300, Route 9
Fishkill 12524
Tel: (914) 897-3860
FAX: (914) 897-3125

NORTH CAROLINA

†Intel Corp
5800 Executive Center Dr
Suite 105
Charlotte 28212
Tel: (704) 568-8966
FAX: (704) 535-2236

Intel Corp
5540 Centerview Dr
Suite 215
Raleigh 27606
Tel: (919) 851-9537
FAX: (919) 851-8974

OHIO

†Intel Corp.*
3401 Park Center Drive
Suite 220
Dayton 45414
Tel: (513) 890-5350
TWX: 810-450-2528
FAX: (513) 890-8658

†Intel Corp.*
25700 Science Park Dr
Suite 100
Beachwood 44122
Tel: (216) 464-2736
TWX: 810-427-9298
FAX: (804) 282-0673

OKLAHOMA

Intel Corp
6801 N. Broadway
Suite 115
Oklahoma City 73162
Tel: (405) 848-8086
FAX: (405) 840-9819

OREGON

†Intel Corp
15254 N.W. Greenbrier Parkway
Building B
Beaverton 97005
Tel: (503) 645-8051
TWX: 910-467-8741
FAX: (503) 645-8181

PENNSYLVANIA

†Intel Corp.*
455 Pennsylvania Avenue
Suite 230
Fort Washington 19034
Tel: (215) 641-1000
TWX: 510-661-2077
FAX: (215) 641-0785

†Intel Corp.*
400 Penn Center Blvd
Suite 610
Pittsburgh 15235
Tel: (412) 823-4970
FAX: (412) 829-7578

PUERTO RICO

†Intel Corp
South Industrial Park
P.O. Box 910
Las Piedras 00671
Tel: (809) 733-8616

TEXAS

Intel Corp
8911 Capital of Texas Hwy
Austin 78759
Tel: (512) 794-8086
FAX: (512) 338-9335

†Intel Corp.*
12000 Ford Road
Suite 400
Dallas 75234
Tel: (214) 241-8087
FAX: (214) 484-1180

†Intel Corp.*
7322 S.W. Freeway
Suite 1490
Houston 77074
Tel: (713) 988-8086
TWX: 910-881-2490
FAX: (713) 988-3660

UTAH

†Intel Corp
428 East 6400 South
Suite 104
Murray 84107
Tel: (801) 263-8051
FAX: (801) 268-1457

VIRGINIA

†Intel Corp
1504 Santa Rosa Road
Suite 108
Richmond 23288
Tel: (804) 282-5668
FAX: (216) 464-2270

WASHINGTON

†Intel Corp
155 108th Avenue N.E.
Suite 386
Bellevue 98004
Tel: (206) 453-8086
TWX: 910-443-3002
FAX: (206) 451-9556

Intel Corp.
408 N. Mullan Road
Suite 102
Spokane 99206
Tel: (509) 928-8086
FAX: (509) 928-9467

WISCONSIN

Intel Corp
330 S. Executive Dr
Suite 102
Brookfield 53005
Tel: (414) 784-8087
FAX: (414) 796-2115

CANADA

BRITISH COLUMBIA

Intel Semiconductor of Canada, Ltd.
4585 Canada Way
Suite 202
Burnaby V5G 4L6
Tel: (604) 298-0387
FAX: (604) 298-8234

ONTARIO

†Intel Semiconductor of Canada, Ltd.
2650 Queensview Drive
Suite 250
Ottawa K2B 8H6
Tel: (613) 829-9714
FAX: (613) 820-5936

†Intel Semiconductor of Canada, Ltd.
190 Attwell Drive
Suite 500
Rexdale M9W 6H8
Tel: (416) 675-2105
FAX: (416) 675-2438

QUEBEC

Intel Semiconductor of Canada, Ltd.
620 St. Jean Boulevard
Pointe Claire H9R 3K2
Tel: (514) 694-9130
FAX: 514-694-0064

†Sales and Service Office
*Field Application Location

intel

EUROPEAN SALES OFFICES

DENMARK

Intel Denmark A/S
Glentevej 61 3rd Floor
2400 Copenhagen NV
Tel (45) (31) 19 80 33
TLX 19567

FINLAND

Intel Finland OY
Ruosilantie 2
00390 Helsinki
Tel (358) 0 544 644
TLX 123332

FRANCE

Intel Corporation S A R L
1 Rue Edison BP 303
78054 St Quentin en Yvelines
Cedex
Tel (33) (1) 30 57 70 00
TLX 699016

WEST GERMANY

Intel Semiconductor GmbH*
Dornacher Strasse 1
8016 Feldkirchen bei Muenchen
Tel (49) 089 90992-0
TLX 5-23177

Intel Semiconductor GmbH
Hohenzollern Strasse 5
3000 Hannover 1
Tel (49) 0511 344081
TLX 9 23625

Intel Semiconductor GmbH
Abraham Lincoln Strasse 16 18
6200 Wiesbaden
Tel (49) 06121 7605-0
TLX 4-186183

Intel Semiconductor GmbH
Zettachring 10A
7000 Stuttgart 80
Tel (49) 0711 7287 280
TLX 7 254826

ISRAEL

Intel Semiconductor Ltd *
Atidim Industrial Park-Neve Sharet
P O Box 43202
Tel Aviv 61430
Tel (972) 3 548 3222
TLX 371215

ITALY

Intel Corporation Italia S p A *
Milanofiori Palazzo E
20090 Assago
Milano
Tel (39) (02) 89200950
TLX 341286

NETHERLANDS

Intel Semiconductor B V *
Postbus 84130
3099 CC Rotterdam
Tel (31) 10 407 11 11
TLX 22283

NORWAY

Intel Norway A/S
Hvamveien 4 PO Box 92
2013 Skjetten
Tel (47) (6) 842 420
TLX 78018

SPAIN

Intel Iberia S A
Zurbaran 28
28010 Madrid
Tel (34) (1) 308 25 52
TLX 46880

SWEDEN

Intel Sweden A B *
Dalvagen 24
171 36 Solna
Tel (46) 8 734 01 00
TLX 12261

SWITZERLAND

Intel Semiconductor A G
Zuerichstrasse
8185 Winkel Rueti bei Zuerich
Tel (41) 01/860 62 62
TLX 825977

UNITED KINGDOM

Intel Corporation (U K) Ltd *
Pipers Way
Swindon, Wiltshire SN3 1RJ
Tel (44) (0793) 696000
TLX 444447/8

EUROPEAN DISTRIBUTORS/REPRESENTATIVES

AUSTRIA

Bacher Electronics G m b H
Rotenmuehlgasse 26
1120 Wien
Tel (43) (0222) 83 56 46
TLX 31532

BELGIUM

Inelco Belgium S A
Av des Croix de Guerre 94
1120 Bruxelles
Oorlogskruisenlaan 94
1120 Brussel
Tel (32) (02) 216 01 60
TLX 64475 or 22090

DENMARK

ITT Multikomponent
Naverland 29
2600 Glostrup
Tel (45) (0) 2 45 66 45
TLX 33 355

FINLAND

OY Fintronic AB
Melkonkatu 24A
00210 Helsinki
Tel (358) (0) 6926022
TLX 124224

FRANCE

Almex
Zone industrielle d'Antony
48, rue de l Aubepine
BP 102
92164 Antony cedex
Tel (33) (1) 46 66 21 12
TLX 250067

Jermyn-Generim
60, rue des Gemeaux
Silic 580
94653 Rungis cedex
Tel (33) (1) 49 78 49 78
TLX 261585

Metrologie
Tour d Asnieres
4, av Laurent-Cely
92606 Asnieres Cedex
Tel (33) (1) 47 90 62 40
TLX 611448

Tekelec Airtronic
Cite des Bruyeres
Rue Carle Vernet BP 2
92310 Sevres
Tel (33) (1) 45 34 75 35
TLX 204552

WEST GERMANY

Electronic 2000 AG
Stahlgruberring 12
8000 Muenchen 82
Tel (49) 089 42001-0
TLX 522561

ITT Multikomponent GmbH
Postfach 1265
Bahnhofstrasse 44
7141 Moeglingen
Tel (49) 07141 4879
TLX 7264472

Jermyn GmbH
Im Dachsstueck 9
6250 Limburg
Tel (49) 06431 508 0
TLX 415257 0

Metrologie GmbH
Meglingerstrasse 49
8000 Muenchen 71
Tel (49) 089 78042 0
TLX 5213189

Proelectron Vertriebs GmbH
Max Planck Strasse 1-3
6072 Dreieich
Tel (49) 06103.30434 3
TLX 417903

IRELAND

Micro Marketing Ltd
Glenageary Office Park
Glenageary
Co Dublin
Tel (21) (353) (01) 85 63 25
TLX 31584

ISRAEL

Eastronics Ltd
11 Rozanis Street
P O B 39300
Tel-Aviv 61392
Tel (972) 03 475151
TLX 33638

ITALY

Intesi
Divisione ITT Industries GmbH
Viale Milanofiori
Palazzo E 5
20090 Assago (MI)
Tel (39) 02 824701
TLX 311351

Lasi Elettronica S p A
V le Fulvio Testi 126
20092 Cinisello Balsamo (MI)
Tel (39) 02 2440012
TLX 352040

Telcom S r l
Via M Civitali 75
20148 Milano
Tel (39) 02 4049046
TLX 335654

ITT Multicomponents
Viale Milanofiori E 5
20090 Assago (MI)
Tel (39) 02 824701
TLX 311351

Silverstar
Via Dei Gracchi 20
20146 Milano
Tel (39) 02 49961
TLX 332189

NETHERLANDS

Koning en Hartman Elektrotechniek B V
Energieweg 1
2627 AP Delft
Tel (31) (0) 15 609906
TLX 38250

NORWAY

Nordisk Elektronikk (Norge) A S
Postboks 123
Smedsvingen 4
1364 Hvalstad
Tel (47) (02) 84 62 10
TLX 77546

PORTUGAL

ATD Portugal LDA
Rua Dos Lusiados 5 Sala B
1300 Lisboa
Tel (35) (1) 64 80 91
TLX 61562

Ditram
Avenida Miguel Bombarda, 133
1000 Lisboa
Tel (35) (1) 54 53 13
TLX 14182

SPAIN

ATD Electronica S A
Plaza Ciudad de Viena 6
28040 Madrid
Tel (34) (1) 234 40 00
TLX 42477

ITT-SESA
Calle Miguel Angel, 21-3
28010 Madrid
Tel (34) (1) 419 09 57
TLX 27461

Metrologia Iberica S A
Ctra de Fuencarral n 80
28100 Alcobendas (Madrid)
Tel (34) (1) 653 86 11

SWEDEN

Nordisk Elektronik AB
Torshamnsgatan 39
Box 36
164 93 Kista
Tel (46) 08-03 46 30
TLX 105 47

SWITZERLAND

Industrade A G
Hertistrasse 31
8304 Wallisellen
Tel (41) (0) 1 8328111
TLX 56788

TURKEY

EMPA Electronic
Lindwurmstrasse 95A
8000 Muenchen 2
Tel (49) 089/53 80 570
TLX 528573

UNITED KINGDOM

Accent Electronic Components Ltd
Jubilee House, Jubilee Road
Letchworth Herts SG6 1TL
Tel (44) (0462) 686666
TLX 826293

Bytech Comway Systems
3 The Western Centre
Western Road
Bracknell RG12 1RW
Tel (44) (0344) 55333
TLX 847201

Jermyn
Vestry Estate
Otford Road
Sevenoaks
Kent TN14 5EU
Tel (44) (0732) 450144
TLX 95142

MMD
Unit 8 Southview Park
Caversham
Reading
Berkshire RG4 0AF
Tel (44) (0734) 481666
TLX 846669

Rapid Silicon
Rapid House
Denmark Street
High Wycombe
Buckinghamshire HP11 2ER
Tel (44) (0494) 442266
TLX 837931

Rapid Systems
Rapid House
Denmark Street
High Wycombe
Buckinghamshire HP11 2ER
Tel (44) (0494) 450244
TLX 837931

YUGOSLAVIA

H R Microelectronics Corp
2005 de la Cruz Blvd , Ste 223
Santa Clara, CA 95050
U S A
Tel (1) (408) 988 0286
TLX 387452

Rapido Electronic Components
S p a
Via C Beccaria, 8
34133 Trieste
Italia
Tel (39) 040/360555
TLX 460461

*Field Application Location

intel®

INTERNATIONAL SALES OFFICES

AUSTRALIA

Intel Australia Pty. Ltd. *
Spectrum Building
200 Pacific Hwy., Level 6
Crows Nest, NSE, 2065
Tel: 612-957-2744
FAX: 612-923-2632

BRAZIL

Intel Semicondutores do Brazil LTDA
Av. Paulista, 1159-CJS 404/405
01311 - Sao Paulo - S.P.
Tel: 55-11-287-5899
TLX: 3911153146 ISDB
FAX: 55-11-287-5119

CHINA/HONG KONG

Intel PRC Corporation
15/F, Office 1, Citic Bldg.
Jian Guo Men Wai Street
Beijing, PRC
Tel: (1) 500-4850
TLX: 22947 INTEL CN
FAX: (1) 500-2953

Intel Semiconductor Ltd. *
10/F East Tower
Bond Center
Queensway, Central
Hong Kong
Tel: (852) 844-4555
FAX: (852) 868-1989

INDIA

Intel Asia Electronics, Inc.
4/2, Samrah Plaza
St. Mark's Road
Bangalore 560001
Tel: 011-91-812-215065
TLX: 9538452875 DCBY
FAX: 091-812-215067

JAPAN

Intel Japan K.K.
5-6 Tokodai, Tsukuba-shi
Ibaraki, 300-26
Tel: 0298-47-8511
TLX: 3656-160
FAX: 0298-47-8450

Intel Japan K.K. *
Daiichi Mitsugi Bldg.
1-8889 Fuchu-cho
Fuchu-shi, Tokyo 183
Tel: 0423-60-7871
FAX: 0423-60-0315

Intel Japan K.K. *
Bldg. Kumagaya
2-69 Hon-cho
Kumagaya-shi, Saitama 360
Tel: 0485-24-6871
FAX: 0485-24-7518

Intel Japan K.K. *
Mitsui-Seimei Musashi-kosugi Bldg.
915 Shinmaruko, Nakahara-ku
Kawasaki-shi, Kanagawa 211
Tel: 044-733-7011
FAX: 044-733-7010

Intel Japan K.K.
Nihon Seimei Atsugi Bldg.
1-2-1 Asahi-machi
Atsugi-shi, Kanagawa 243
Tel: 0462-29-3731
FAX: 0462-29-3781

Intel Japan K.K. *
Ryokuchi-Eki Bldg.
2-4-1 Terauchi
Toyonaka-shi, Osaka 560
Tel: 06-863-1091
FAX: 06-863-1084

Intel Japan K.K.
Shinmaru Bldg.
1-5-1 Marunouchi
Chiyoda-ku, Tokyo 100
Tel: 03-201-3621
FAX: 03-201-6850

Intel Japan K.K.
Green Bldg.
1-16-20 Nishiki
Naka-ku, Nagoya-shi,
Aichi 450
Tel: 052-204-1261
FAX: 052-204-1285

KOREA

Intel Technology Asia, Ltd.
16th Floor, Life Bldg.
61 Yoido-dong, Youngdeungpo-Ku
Seoul 150-010
Tel: (2) 784-8186, 8286, 8386
TLX: K29312 INTELKO
FAX: (2) 784-8096

SINGAPORE

Intel Singapore Technology, Ltd.
101 Thomson Road #21-05/06
United Square
Singapore 1130
Tel: 250-7811
TLX: 39921 INTEL
FAX: 250-9256

TAIWAN

Intel Technology Far East Ltd.
8th Floor, No. 205
Bank Tower Bldg.
Tung Hua N. Road
Taipei
Tel: 886-2-716-9660
FAX: 886-2-717-2455

INTERNATIONAL DISTRIBUTORS/REPRESENTATIVES

ARGENTINA

Dafsys S.R.L.
Chacabuco, 90-6 Piso
1069-Buenos Aires
Tel: 54-1-334-7726
FAX: 54-1-334-1871

AUSTRALIA

Email Electronics
15-17 Hume Street
Huntingdale, 3166
Tel: 011-61-3-544-8244
TLX: AA 30895
FAX: 011-61-3-543-8179

NSD-Australia
205 Middleborough Rd.
Box Hill, Victoria 3128
Tel: 03 8900970
FAX: 03 8990819

BRAZIL

Elebra Microelectronica S.A.
Rua Geraldo Flausina Gomes, 78
7 Andar
04575 - Sao Paulo - S.P.
Tel: 55-11-534-9641
TLX: 55-11-54593/54591
FAX: 55-11-534-9424

CHINA/HONG KONG

Novel Precision Machinery Co., Ltd.
Flat D, 20 Kingsford Ind. Bldg.
Phase 1, 26 Kwai Hei Street
N.T., Kowloon
Hong Kong
Tel: (852) 422-3222
TWX: 39114 JINMI HX
FAX: (852) 426-1602

INDIA

Micronic Devices
Arun Complex
No. 65 D.V.G. Road
Basavanagudi
Bangalore 560 004
Tel: 011-91-812-600-631
011-91-812-611-365
TLX: 9538458332 MDBG

Micronic Devices
No. 516 5th Floor
Swastik Chambers
Sion, Trombay Road
Chembur
Bombay 400 071
TLX: 9531 171447 MDEV

Micronic Devices
25/8, 1st Floor
Bada Bazaar Marg
Old Rajinder Nagar
New Delhi 110 060
Tel: 011-91-11-5723509
011-91-11-589771
TLX: 031-63253 MDND IN

Micronic Devices
6-3-348/12A Dwarakapuri Colony
Hyderabad 500 482
Tel: 011-91-842-226748

S&S Corporation
1587 Kooser Road
San Jose, CA 95118
Tel: (408) 978-6216
TLX: 820281
FAX: (408) 978-8635

JAPAN

Asahi Electronics Co. Ltd.
KMM Bldg. 2-14-1 Asano
Kokurakita-ku
Kitakyushu-shi 802
Tel: 093-511-6471
FAX: 093-551-7861

C. Itoh Techno-Science Co., Ltd.
4-8-1 Dobashi, Miyamae-ku
Kawasaki-shi, Kanagawa 213
Tel: 044-852-5121
FAX: 044-877-4268

Dia Semicon Systems, Inc.
Flower Hill Shinmachi Higashi-kan
1-23-9 Shinmachi, Setagaya-ku
Tokyo 154
Tel: 03-439-1600
FAX: 03-439-1601

Okaya Koki
2-4-18 Sakae
Naka-ku, Nagoya-shi 460
Tel: 052-204-2916
FAX: 052-204-2901

Ryoyo Electro Corp.
Konwa Bldg.
1-12-22 Tsukiji
Chuo-ku, Tokyo 104
Tel: 03-546-5011
FAX: 03-546-5044

KOREA

J-Tek Corporation
6th Floor, Government Pension Bldg.
24-3 Yoido-dong
Youngdeungpo-ku
Seoul 150-010
Tel: 82-2-780-8039
TLX: 25299 KODIGIT
FAX: 82-2-784-8391

Samsung Electronics
150 Taepyungro-2 KA
Chungku, Seoul 100-102
Tel: 82-2-751-3985
TLX: 27970 KORSST
FAX: 82-2-753-0967

MEXICO

SSB Electronics, Inc.
675 Palomar Street, Bldg. 4, Suite A
Chula Vista, CA 92011
Tel: (619) 585-3253
TLX: 287751 CBALL UR
FAX: (619) 585-8322

Dicopel S.A.
Tocnti 368 Fracc. Ind. San Antonio
Azcapotzalco
C.P. 02760-Mexico, D.F.
Tel: 52-5-561-3211
TLX: 177 3790 Dicome
FAX: 52-5-561-1279

PSI S.A. de C.V.
Fco. Villa esq. Ajusco s/n
Cuernavaca - Morelos
Tel: 52-73-13-9412
FAX: 52-73-17-5333

NEW ZEALAND

Email Electronics
36 Olive Road
Penrose, Auckland
Tel: 011-64-9-591-155
FAX: 011-64-9-592-681

SINGAPORE

Electronic Resources Pte. Ltd.
17 Harvey Road #04-01
Singapore 1336
Tel: 283-0888
TWX: 56541 ERS
FAX: 2895327

SOUTH AFRICA

Electronic Building Elements
178 Erasmus Street (off Watermeyer Street)
Meyerspark, Pretoria, 0184
Tel: 011-2712-803-7680
FAX: 011-2712-803-8294

TAIWAN

Micro Electronics Corporation
5/F 587, Ming Shen East Rd.
Taipei, R.O.C.
Tel: 886-2-501-8231
FAX: 886-2-505-6609

Sertek
15/F 135, Section 2
Chien Juo North Rd.
Taipei 10479, R.O.C.
Tel: (02) 5010055
FAX: (02) 5012521
(02) 5058414

VENEZUELA

P. Benavides S.A.
Avilanes a Rio
Residencia Kamarata
Locales 4 AL 7
La Candelaria, Caracas
Tel: 58-2-574-6338
TLX: 28450
FAX: 58-2-572-3321

*Field Application Location

UNITED STATES, Intel Corporation
3065 Bowers Ave., Santa Clara, CA 95051
Tel: (408) 765-8080

JAPAN, INTEL Japan K.K.
5-6 Tokodai, Tsukuba-shi, Ibaraki, 300-26
Tel: 0298-47-8511

FRANCE, Intel Corporation S.A.R.L.
1, Rue Edison, BP 303, 78054 Saint-Quentin-en-Yvelines Cedex
Tel: (33) (1) 30 57 70 00

UNITED KINGDOM, Intel Corporation (U.K.) Ltd.
Pipers Way, Swindon, Wiltshire, England SN3 1RJ
Tel: (44) (0793) 696000

WEST GERMANY, Intel Semiconductor GmbH
Dornacher Strasse 1
8016 Feldkirchen bei Muenchen
Tel: (49) 089/90992-0

HONG KONG, Intel Semiconductor Ltd.
10/F East Tower, Bond Center, Queensway, Central
Tel: (852) 844-4555

CANADA, Intel Semiconductor of Canada, Ltd.
190 Attwell Drive, Suite 500
Rexdale, Ontario M9W 6H8
Tel: (416) 675-2105

Printed in U.S.A./TP491/0590/4M/PE JB
Embedded Controllers

CG/011890